Yee & Associates, P.C.

4100 Alpha Road
Suite 1100
Dallas, Texas 75244

Main No. (972) 385-8777
Facsimile (972) 385-7766

RECEIVED
CENTRAL FAX CENTER

MAR 2 1 2005

Facsimile Cover Sheet

To: Commissioner for Patents for **Examiner Jeffrey R. West** **Group Art Unit 2857**	Facsimile No.: **703/872-9306**
From: Lourdes Perez for Carrie Parker Legal Assistant to Cathrine K. Kinslow/jl	No. of Pages Including Cover Sheet: **34**

Message:

Enclosed herewith:

- Transmittal Document; and
- Appeal Brief.

Re: Application No. **09/871,498**
Attorney Docket No: **RSW920010060US1**

Date: Monday, March 21, 2005

Please contact us at (972) 385-8777 if you do not receive all pages indicated above or experience any difficulty in receiving this facsimile.

PLEASE CONFIRM RECEIPT OF THIS TRANSMISSION BY FAXING A CONFIRMATION TO 972-385-7766.

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re application of: **Etgen**

Serial No.: **09/871,498**

Filed: **May 31, 2001**

For: **Method and Apparatus for Calculating Data Integrity Metrics for Web Server Activity Log Analysis**

Group Art Unit: **2857**

Examiner: **West, Jeffrey R.**

Attorney Docket No.: **RSW920010060US1**

Certificate of Transmission Under 37 C.F.R. § 1.8(a)
I hereby certify this correspondence is being transmitted via facsimile to the Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450, facsimile number (703) 872-9306 on March 21, 2005.

By: Lourdes Perez

36736
PATENT TRADEMARK OFFICE
CUSTOMER NUMBER

TRANSMITTAL DOCUMENT

Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Sir:
ENCLOSED HEREWITH:

- Appeal Brief (37 C.F.R. 41.37).

A fee of $500.00 is required for filing an Appeal Brief. Please charge this fee to IBM Corporation Deposit Account No. 09-0461. No additional fees are believed to be necessary. If, however, any additional fees are required, I authorize the Commissioner to charge these fees which may be required to IBM Corporation Deposit Account No. 09-0461. No extension of time is believed to be necessary. If, however, an extension of time is required, the extension is requested, and I authorize the Commissioner to charge any fees for this extension to IBM Corporation Deposit Account No. 09-0461.

Respectfully submitted,

Cathrine K. Kinslow
Registration No. 51,886
Duke W. Yee
Registration No. 34,285
YEE & ASSOCIATES, P.C.
P.O. Box 802333
Dallas, Texas 75380
(972) 385-8777
ATTORNEYS FOR APPLICANT

RECEIVED
CENTRAL FAX CENTER
MAR 2 1 2005

Docket No. RSW920010060US1 *PATENT*

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re application of: **Etgen**	§	
	§	Group Art Unit: **2857**
Serial No. **09/871,498**	§	
	§	Examiner: **West, Jeffrey R.**
Filed: **May 31, 2001**	§	
	§	
For: **Method and Apparatus for Calculating Data Integrity Metrics for Web Server Activity Log Analysis**	§	Attorney Docket No.: **RSW920010060US1**

Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Certificate of Transmission Under 37 C.F.R. § 1.8(a)
I hereby certify this correspondence is being transmitted via facsimile to the Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450, facsimile number (703) 872-9306 on March 21, 2005.
By: Lourdes Perez

APPEAL BRIEF (37 C.F.R. 41.37)

This brief is in furtherance of the Notice of Appeal, filed in this case on January 19, 2005.

The fees required under § 41.20(B)(2), and any required petition for extension of time for filing this brief and fees therefore, are dealt with in the accompanying TRANSMITTAL OF APPEAL BRIEF.

REAL PARTIES IN INTEREST

The real party in interest in this appeal is the following party: International Business Machines Corporation.

RELATED APPEALS AND INTERFERENCES

With respect to other appeals or interferences that will directly affect, or be directly affected by, or have a bearing on the Board's decision in the pending appeal, there are no such appeals or interferences.

STATUS OF CLAIMS

A. TOTAL NUMBER OF CLAIMS IN APPLICATION

Claims in the application are: 1, 3-16, 18-29, and 31-41

B. STATUS OF ALL THE CLAIMS IN APPLICATION

1. Claims canceled: 2, 17, and 30
2. Claims withdrawn from consideration but not canceled: NONE
3. Claims pending: 1, 3-16, 18-29, and 31-41
4. Claims allowed: NONE
5. Claims rejected: 1, 3-16, 18-29, and 31-41
6. Claims objected to: NONE.

C. CLAIMS ON APPEAL

The claims on appeal are: 1, 3-16, 18-29, and 31-41.

STATUS OF AMENDMENTS

There are no amendments after final rejection.

SUMMARY OF CLAIMED SUBJECT MATTER

A. CLAIMS 1, 14, and 29 - INDEPENDENT

The present invention provides a method in a data processing system for maintaining data integrity in logs (Specification, page 12, lines 1-4), the method comprising: reviewing a log, wherein the log includes a set of time segments, and wherein the set of time segments include at least one time segment defined by a user (Specification, page 12, line 17, to page 13, line 2, Figure 3, log analyzer 300, log 302); determining whether the log contains a data loss, wherein the determining step includes analyzing each time segment within the set of time segments to determine whether a time segment gap tolerance has been exceeded (Specification, page 16, lines 12-17, Figure 5, step 502); and adding data to replace the data loss in the log to increase integrity of the log if a determination is made that a data loss has occurred (Specification, page 16, lines 20-21, Figure 5, step 508). These features are recited in independent claims 1, 14, and 29 of the present invention.

B. CLAIMS 8, 15, and 36 - INDEPENDENT

The present invention also provides a method in a data processing system for analyzing a log (Specification, page 12, lines 1-4), the method comprising: analyzing a set of time segments in the log to determine whether a time gap tolerance has been exceeded for a time segment within the set of time segments, wherein the set of time segments include at least one time segment defined by a user (Specification, page 12, line 17, to page 13, line 2, Figure 3, log analyzer 300, log 302); and responsive to a determination that the time gap tolerance has been exceeded for the time segment within the set of time segments, generating an alert (Specification, page 16, lines 17-18, Figure 5, step 504). These features are recited in independent claims 8, 15, and 36 of the present invention.

C. CLAIM 16 - INDEPENDENT

Furthermore, the present invention provides a data processing system for maintaining data integrity in logs (Specification, page 12, lines 1-4), the data processing system comprising: reviewing means for reviewing a log, wherein the log includes a set of time segments, and wherein the set of time segments include at least one time segment defined by a user (Specification, page 12, line 17, to page 13, line 2, Figure 3, log analyzer 300); determining means for determining whether the log contains a data loss by analyzing each time segment

within the set of time segments to determine whether a time segment gap tolerance has been exceeded (Specification, page 16, lines 10-17, Figure 3, log analyzer 300); and adding means for adding data to replace the data loss in the log to increase integrity of the log if a determination is made that a data loss has occurred (Specification, page 16, lines 10-21, Figure 3, log analyzer 300). These features are recited in independent claim 16 of the present invention.

D. CLAIM 21 - DEPENDENT

Moreover, the present invention provides a data processing system (Specification, page 12, lines 1-4),wherein the analyzing means includes considering data in at least one time segment adjacent to a time segment being analyzed (Specification, page 17, line 12, to page 18, line 10, Figure 3, log analyzer, Figure 6, step 600). These features are recited in claim 21 of the present invention.

E. CLAIM 23 - INDEPENDENT

Additionally, the present invention provides a data processing system for analyzing a log (Specification, page 12, lines 1-4), the data processing system comprising: analyzing means for analyzing a set of time segments in the log to determine whether a time gap tolerance has been exceeded for a time segment within the set of time segments, wherein the set of time segments include at least one time segment defined by a user (Specification, page 20, line 25, to page 21, line 5, Figure 3, log analyzer 300, Figure 7, step 704); and generating means, responsive to a determination that the time gap tolerance has been exceeded for the time segment within the set of time segments, for generating an alert (Specification, page 16, lines 10-18, Figure 3, log analyzer 300, Figure 5, step 504). These features are recited in independent claim 23 of the present invention.

F. CLAIM 24 - DEPENDENT

In addition, the present invention provides a data processing system (Specification, page 12, lines 1-4) further comprising: adding means, responsive to detecting the alert, for adding data to the time segment within the set of time segments to increase the data integrity of the log (Specification, page 16, lines 10-21, Figure 3, log analyzer 300). These features are recited in claim 24 of the present invention.

G. CLAIM 27 - DEPENDENT

What is more, the present invention provides a data processing system (Specification, page 12, lines 1-4) further comprising: calculating means for calculating a data integrity level for

the log (Specification, page 20, line 25, to page 21, line 17, Figure 3, log analyzer 300, Figure 7, step 712). These features are recited in claim 27 of the present invention.

36, reads as follows:

> 8. A method in a data processing system for analyzing a log, the method comprising:
>
> analyzing a set of time segments in the log to determine whether a time gap tolerance has been exceeded for a time segment within the set of time segments, wherein the set of time segments include at least one time segment defined by a user; and
>
> responsive to a determination that the time gap tolerance has been exceeded for the time segment within the set of time segments, generating an alert.

Claim 8 recites the feature of analyzing a set of time segments in a log to determine whether a time gap tolerance has been exceeded for a time segment within the set of time segments. The Examiner cites the following sections of *Wu* as teaching this analyzing feature:

> The time gap represents the difference between the time stamp of the current log and that of the access pair in the tail of the active session.

(*Wu*, column 7, lines 36-39).

> FIG. 8 shows an example of a method for finding an active candidate session to append the current log entry and marking all the sessions which are considered dormant and should be closed (step 309). As depicted, in step 801, the first active session is selected from the head of the list of the active sessions. The process ends, in step 804, when all active sessions have been processed. In step 805, If the time gap is greater than a pre-specified time, TIME_FOR_CLOSE, for closing a session, then the dormant flag in the session header is set to a true value, in step 806.

(*Wu*, column 9, lines 57-66).

The passages above teach a time gap, which represents the difference between the timestamp of the current log and the timestamp of the access pair in the tail of an active session. The passages also teach finding an active session to append the current log entry and marking sessions considered dormant. However, neither passage above teaches analyzing a set of time segments in a log to determine whether a time gap tolerance has been exceeded for a time segment, as taught by claim 8 of the present invention.

Evaluating the superficial similarities between *Wu* and the present invention highlights the evident differences between the two. *Wu* teaches time stamps, time gaps between time stamps, and whether or not time gaps are greater than pre-specified times, but *Wu* does not teach

time segments. On the other hand, the present invention teaches time segments, time gap tolerances, and whether or not time gap tolerances have been exceeded for time segments. While *Wu* teaches a time gap and the present invention teaches a somewhat similar time gap tolerance, how the time gaps are applied is quite different. *Wu*'s time gap is a singular calculation, the difference between a current log's time stamp and an access pair's time stamp. In contrast, the present invention provides a log's time segment, with a starting time and an ending time, which is analyzed numerous times for time gap tolerances. An example is illustrated in Figure 4B of the application, where a five hour time segment in a log is analyzed using a time gap tolerance of 8,200 milliseconds, which would result in over 2,000 comparisons, quite different than *Wu*'s singular comparison between the time stamps of a log and an access pair.

The section of *Wu* that the Examiner cites as teaching a time segment only references the previously mentioned time gap again. Reviewing the definition of a time segment makes obvious the absence of a time segment in *Wu*. A time segment is a period of time, with both a starting time and an ending time, such as a period of hours in a day. Neither the current log nor the access pair in *Wu* has a starting time and an ending time, for such concepts do not apply to either feature in *Wu*. The log in *Wu* does not represent a period of time, nor does the access pair in *Wu*. Both the log and the access pair in *Wu* have a time stamp, but a time stamp is not a time segment because a time stamp is a single data point that represents a single point in time. A time stamp is not a period of time with both a starting time and an ending time. Thus, *Wu* teaches analyzing a time gap between two time stamps, rather than analyzing a set of time segments in a log to determine whether a time gap tolerance has been exceeded for a time segment, as recited in claim 8 of the present invention.

All limitations of the claimed invention must be considered when determining patentability. *In re Lowry*, 32 F.3d 1579, 1582, 32 U.S.P.Q.2d 1031, 1034 (Fed. Cir. 1994). In comparing *Wu* to the claimed invention to determine obviousness, limitations of the presently claimed invention may not be ignored. The present invention in claim 8 recites analyzing a set of time segments in a log to determine whether a time gap tolerance has been exceeded for a time segment. Such a feature is not taught or suggested by *Wu*. Therefore, claim 8 is not obvious in view of *Wu*.

Furthermore, there is no teaching or suggestion in either *Wu* or *Boyd* as to the desirability of including the features from the other reference. Although the Examiner relies upon *Boyd* to teach a time segment defined by a user, the purpose of *Boyd's* one-pass routine is to "summarize

the access information recorded for a user-requested time frame," which "enables the server 10 to efficiently analyze traffic data by utilizing existing summaries 19A-C whenever possible." (Column 9, lines 17-47). *Boyd* does not even mention time gaps or time gap tolerance, much less the need for detecting a time gap, as specified in the present invention. This is because *Boyd* is concerned with summarizing data previously existing for time slices, not with the present invention's detection of time gaps in logs in order to determine which logs to process differently. Because the concept of a time gap is inapplicable in *Boyd*, there is no teaching or suggestion to modify *Boyd* to include the teachings of inventions addressing the issue of time gaps, such as in *Wu*.

Additionally, *Wu's* use of time gaps also differs significantly from *Boyd's* use of time slices. This is because *Wu* uses time gaps in "[a] method and system to map client access patterns." (Abstract) *Wu*'s use of time gaps contrasts with *Boyd*'s summarizing data previously existing for time slices. Because, as discussed above, the concept of a time segment in inapplicable in *Wu*, there is no teaching or suggestion to modify *Wu* to include the teachings of inventions summarizing data previously existing for time segments, such as in *Boyd*. Understandably, there is no teaching or suggestion in either *Wu* or *Boyd* as to the desirability of including the features from the other reference.

Therefore, even if *Wu* taught analyzing a set of time segments in a log to determine whether a time gap tolerance has been exceeded for a time segment, as alleged by the Examiner, the combination of *Wu* with *Boyd* would still not be the invention as recited in claim 8 of the present invention. Rather, such an alleged combination would result in a system using time gaps for mapping client access patterns, substantially as taught in *Wu*, with the additional feature of summarizing data previously existing for time slices, substantially as taught in *Boyd*. A system that uses time gaps in relation to access pairs in the tail of an active session is both real-time and dynamic. Such a system would not benefit from the addition of a feature oriented towards summarizing historical and static data. Even considering *Boyd*, the cited references fail to teach or suggest analyzing a set of time segments in a log to determine whether a time gap tolerance has been exceeded for a time segment, as recited in claim 8.

Moreover, even if *Wu* taught analyzing a set of time segments in a log to determine whether a time gap tolerance has been exceeded for a time segment, as alleged by the Examiner, one of ordinary skill in the art would not combine *Wu* with *Boyd* when each reference is

considered as a whole. In considering the references as a whole, one of ordinary skill in the art would take into account the problems recognized and solved. For example, *Boyd* teaches:

> . . . a system, method, and storage medium embodying computer-readable code for analyzing traffic data in a distributed computing environment. The distributed computing environment includes a plurality of interconnected systems operatively coupled to a server, a source of traffic data hits and one or more results tables categorized by associated data types.

(Column 2, lines 28-34).

Boyd is directed towards analyzing traffic data in a distributed computing environment through the use of results tables categorized by associated data types. Concerning these results tables, *Boyd* teaches "[e]ach entry in a microtable 45A contains an index 46 pointing to a record within its associated results table 40B which requires adjustment for inflation." (Column7, lines 33-35). *Boyd* explains the need for inflation adjustment through an example: "A count of the number of open sessions spanning each time slice is made in the user session table 43 . . . [d]uring analysis, the access information is adjusted to remove the inflation." (Column 7, lines 46-54). *Boyd* is explicit in how this inflation adjustment is conducted: "The summary of the time frame of the first and second time slice will be inflated unless the double-counts are subtracted from the number of open sessions for this web page for the second time slice." (Column 11, lines 3-6). *Boyd* teaches subtracting counts from summarized results due to "double, triple, or multiple counting of open sessions spanning multiple time slices." (Column 7, lines 61-62). Therefore, *Boyd* is directed towards meeting the "need for a system and method to efficiently process the voluminous amounts of access information generated by web servers in a timely, expedient manner without the attendant costs associated with large scale hardware requirements," by providing a system that "enables the server 10 to efficiently analyze traffic data by utilizing existing summaries 19A-C whenever possible" and subtracts multiple counting to adjust for data inflation. (Column 2, lines 7-11, column 9, lines 45-47, respectively). In summary, *Boyd* efficiently analyzes server traffic through summarizing existing summaries and subtracting inflated counts.

In contrast, *Wu* is directed towards mapping client access patterns in a stateless hypertext server. For each hypertext object access, information about the access is collected, including the hyperlink source (i.e., the hyper-text object that refers the client to the target object) and the hyperlink target (i.e., the hypertext object being accessed). A hyperlink access pair is formed

from the hyperlink source and target, which represents a step in the user traversal path on the hypertext objects. Hyperlink access pairs are mapped into hyperlink access groups. These hyperlink groups can be used to provide user-oriented object usage statistics. (*Wu*, Abstract). Thus, *Wu* teaches mapping client access patterns in order to provide user-oriented object usage statistics and *Boyd* teaches efficiently analyzing server traffic through summarizing existing summaries and subtracting inflated counts. *Wu* is directed towards user-oriented object usage statistics while *Boyd* is directed towards efficiently analyzing server traffic. Therefore, one of ordinary skill in the art would not attempt to combine *Wu* with *Boyd*.

As the Examiner has failed to demonstrate any motivation or incentive in the prior art to combine and modify the references so as to achieve the claimed invention, the alleged combination can only be the result of impermissible hindsight reconstruction using Applicant's own disclosure as a guide. While Applicant understands that all examination entails some measure of hindsight, when the rejection is based completely on hindsight, as in the present case, to the exclusion of what can be gleaned from the references, then the rejection is improper and should be withdrawn.

In view of the above, Applicant submits that independent claims 8, 23, and 36 are not taught or suggested by the alleged combination of *Boyd* and *Wu*. Claims 11, 13, 26, 28, 39, and 41 are dependent claims depending from independent claims 8, 23, and 36, respectively. These dependent claims are also allowable, at least by virtue of their dependency on an allowable claim.

Therefore, the rejection of claims 8, 11, 13, 26, 28, 36, 39, and 41 under 35 U.S.C. § 103 has been overcome.

II. 35 U.S.C. § 103, Obviousness, Claims 1-3, 5-9, 11-18, 20-24, 26-31, 33-37, and 39-41

Claims 1, 3, 5-9, 11-18, 20-24, 26-31, 33-37, and 39-41 are rejected under 35 U.S.C. 103(a) as being unpatentable over *Wu* in view of *Boyd* in further view of *Myerson*. This rejection is respectfully traversed.

With regard to claims 1, 3, 5-9, 11-18, 20-24, 26-31, 33-37, and 39-41, the Examiner states:

> Myerson discloses a method in a data processing system for maintaining data integrity in logs, the method comprising reviewing a log, such as a web-log including request data (column 1, lines 49-52 and column 5, lines 23-54), determining whether the log contains a data loss (column 3, lines 13-17), and

adding data from a prior log to replace the data loss in the log (column 3, lines 17-20) to increase the integrity of the log if a determination is made that a data loss has occurred (column 2, lines 27-42), wherein the data is added to a particular time segment of the log (column 3, lines 13-20).

Myerson discloses that the log includes a sequence/set of time segments (column 2, lines 60-62) calculating a data integrity level for the log and comparing the integrity level to a threshold in order to determine if an acceptable level of integrity has been reached (column 8, lines 47-58).

Myerson also discloses that the method is implemented as a computer program product of corresponding instructions (column 4, line 40) in a system comprising a memory containing the instructions (column 4, lines 18-19), a processing unit for executing the instructions (column 4, lines 16-17), a communications unit (column 4, line 19) and a user interface (column 4, line 19), all connected to a bus (Figure 1).

As noted above, the invention of Myerson teaches many features of the claimed invention, and while Myerson does teach analyzing frequency (i.e. time) difference in the current web-log to determine if data should be appended (column 8, line 66 to column 9, line 7), Myerson does not specifically disclose flagging the determination of an excessive time gap for log analysis. . .

As noted above, the invention of Myerson and Wu teaches many features of the claimed invention, and while invention of Myerson and Wu does disclose analyzing a set of time segments to determine whether a time gap tolerance has been exceeded for a time segment, the combination does not explicitly state that the set of time segments include at least one time segment defined by a user.

Boyd teaches a system and method for analyzing remote traffic data in a distributed computing environment including means for recording a data log of hits (column 2, lines 35-45) with the log segmented into a set of time segments for analysis (column 2, lines 45-51) wherein the set of time segments includes at least one time segment defined by a user (column 9, lines 15-23).

It would have been obvious to one having ordinary skill in the art to modify the invention of Wu to include explicitly stating that the set of time segments include at least one time segment defined by a user, as taught by Boyd, because, as suggested by Boyd, the combination would have allowed the user to select a particular time interval of interest (column 9, lines 50-51) thereby increasing the efficiency of the analysis by only analyzing that which is of interest to the user as well as provided smaller time intervals of analysis to allow greater flexibility and speed in reporting the results (column 9, lines 23-27).

(Final Office Action dated November 22, 2004, pages 4-6).

Independent claim 1, which is representative of independent claims 8, 14, 15, 16, 23, 29, and 36, reads as follows:

1. A method in a data processing system for maintaining data integrity in logs, the method comprising:

> reviewing a log, wherein the log includes a set of time segments, and wherein the set of time segments include at least one time segment defined by a user;
>
> determining whether the log contains a data loss, wherein the determining step includes analyzing each time segment within the set of time segments to determine whether a time segment gap tolerance has been exceeded; and
>
> adding data to replace the data loss in the log to increase integrity of the log if a determination is made that a data loss has occurred.

While *Myerson* teaches a log file (column 2, lines 60-62), Myerson teaches analyzing a web site log file and generating an expanded log file that compensates for information caching and gateway based web site access. The log file used to create the expanded log file contains information representing the Internet address and Authenticated User name of the requester, an Ident name, a timestamp, the request, a response code, and a response size. (Column 5, lines 23-55). *Myerson* does not teach analyzing a set of time segments in a log to determine whether a time gap tolerance has been exceeded for a time segment. *Myerson* merely teaches having timestamp information within the files. This timestamp information specifies the starting time of the communication between the client computer and the web site. (Column 5, lines 43-45). This timestamp information is not a time segment. Thus, while claim 1 of the present invention teaches analyzing a set of time segments in a log to determine whether a time gap tolerance has been exceeded for a time segment, *Myerson* merely teaches automated timestamp data indicating the start of a client session. Therefore, Myerson fails to teach analyzing a set of time segments in a log to determine whether a time gap tolerance has been exceeded for a time segment, as recited in claim 8 of the present invention.

Additionally, claim 1 of the present invention recites the feature of determining whether the log contains a data loss, wherein the determining step includes analyzing each time segment within the set of time segments to determine whether a time segment gap tolerance has been exceeded. The Examiner alleges that Myerson teaches a determination whether a log contains a data loss in the passage below:

> The log expansion procedure furthermore assigns session identifiers to each log record so as to group the sequence of log records into pseudo-client sessions. It then detects when log records are missing from the pseudo-client sessions and supplements the sequence of log records with log records representing additional object requests so as to create pseudo-client sessions each having a logical sequence of object requests.

(*Myerson*, column 3, lines 13-20). The Examiner further states in the Final Office Action that "Myerson does teach analyzing frequency (i.e. time) difference in the current web-log to determine if data should be appended," in the passage below:

> In other words, In one preferred embodiment, if the frequency of requests for the most popular objects is at least 15% lower than the frequency of requests found in the reference request profile, then the threshold amount of object caching for performing additional log entry insertion has been detected. As will be understood by those of ordinary skill in the art, numerous slightly different procedures can be used to compare the current and reference request profiles for the purpose of detecting high rates of object caching.

(*Myerson*, column 8, line 66, to column 9, line 7). As can be seen, neither passage mentions anything about analyzing time segments present in a log to determine whether a time segment gap tolerance has been exceeded. Thus, *Myerson* fails to teach the feature of analyzing each time segment within a set of time segments to determine whether a time segment gap tolerance has been exceeded, as recited in claim 1 of the present invention.

As stated in the argument presented in section I, *Boyd* and *Wu* are not combinable, and *Wu* does not teach all the features that the Examiner alleges. Furthermore, one of ordinary skill in the art would not combine *Boyd* with *Myerson* and *Wu* when each reference is considered as a whole. In considering the references as a whole, one of ordinary skill in the art would take into account the problems recognized and solved. For example, *Myerson* teaches a system and method for analyzing a Web site log file and generating an expanded log file that compensates for information caching and gateway based Web site access. The log expansion procedure analyzes a sequence of log records so as to detect object request patterns indicating that object requests not represented by the log records were satisfied by cached object copies, and then supplements the sequence of log records with inserted log records representing object requests for the objects corresponding to the cached object copies. As a result, the supplemented sequence of log records more accurately represents object requests made by client computers than the initial sequence of log records in the log file. (*Myerson*, Abstract). *Myerson* is directed towards adding data to log files when caching occurs. *Myerson* teaches supplementing a log file to more accurately represent object requests made by a client than the initial sequence of log records in the log file (Abstract). *Myerson* is directed toward adding reasonable data to log files when certain Web content from the content server is cached elsewhere and is therefore not represented

as being accessed in the Web content server log file(s).

In contrast, *Wu* teaches mapping client access patterns in order to provide user-oriented object usage statistics and *Boyd* teaches efficiently analyzing server traffic through summarizing existing summaries and subtracting inflated counts. Thus, *Myerson* is directed towards supplementing log files with cached data, *Wu* is directed towards user-oriented object usage statistics, *Boyd* is directed towards efficiently analyzing server traffic while subtracting inflated counts, and none of the cited references is directed towards supplementing existing log files, as is the present invention.

Furthermore, even if *Myerson* taught what the Examiner alleges, there is no teaching or suggestion in the references as to the desirability of including the features from the other references. As the Examiner has failed to demonstrate any motivation or incentive in the prior art to combine and modify the references so as to achieve the claimed invention, the alleged combination can only be the result of impermissible hindsight reconstruction using Applicant's own disclosure as a guide. While Applicant understands that all examination entails some measure of hindsight, when the rejection is based completely on hindsight, as in the present case, to the exclusion of what can be gleaned from the references, then the rejection is improper and should be withdrawn.

Even if *Wu* was properly combinable with *Boyd* and *Myerson*, the result of such a combination would not be the invention as recited in claim 1 of the present invention. Rather, such an alleged combination would result in a system substantially as taught in *Wu* in addition to the features of summarizing data previously existing for time slices and adding data to log files when caching occurs. Even considering *Myerson*, the cited references fail to teach or suggest analyzing a set of time segments in a log to determine whether a time gap tolerance has been exceeded for a time segment, as recited in claim 1.

In view of the above, Applicant submits that independent claims 1, 8, 14, 15, 16, 23, 29 and 36 are not taught or suggested by the alleged combination of *Boyd*, *Myerson* and *Wu*. Claims 3, 5-7, 9, 11-13, 18, 20-22, 24, 26-28, 31, 33-35, 37, and 39-41 are dependent claims depending from independent claims 1, 8, 16, 23, 29, and 36, respectively. These dependent claims are also allowable, at least by virtue of their dependency on an allowable claim. Therefore, the rejection of claims 1, 3, 5-9, 11-18, 20-24, 26-31, 33-37, and 39-41 under 35 U.S.C. § 103 has been overcome.

III. 35 U.S.C. § 103, Obviousness, Claims 4, 10, 19, 25, 32, and 38

Claims 4, 10, 19, 25, 32 and 38 are rejected under 35 U.S.C. 103(a) as being unpatentable over *Wu* in view of *Boyd* and *Myerson* in further view of *Wilkerson*. This rejection is respectfully traversed.

As stated in the argument presented in section I, *Wu* does not teach all the features that the Examiner alleges, and, as stated in the arguments presented in section II, not only does *Myerson* fail to teach all the features that the Examiner alleges, but *Boyd, Wu* and *Myerson* are not combinable. Furthermore, one of ordinary skill in the art would not combine *Boyd* with *Myerson, Wu* and *Wilkerson* when each reference is considered as a whole. In considering the references as a whole, one of ordinary skill in the art would take into account the problems recognized and solved. For example, *Wilkerson* teaches a method and an apparatus that automates the database recovery process. A person not technically skilled in the use of a computer can operate the new procedure. In accordance with the invention, the user identifies the corrupted database and enters the database name and estimated timestamp for the computer. The database recovery system automatically creates program control language and executes the language wherein the result provides a new, more accurate time stamp. The recovery system then creates additional program control language using the new time stamp and executes the program control language which then recovers the uncorrupted data. The user does not need to look up, write, or enter any program control language because these functions are fully automated. (*Wilkerson*, Abstract). *Wilkerson* is directed towards an automated database recovery process for people who are not technically skilled in the use of a computer. *Wilkerson* teaches the use of estimated timestamps for the purpose of calculating a new, more accurate time stamp, which is then used to recover uncorrupted data. (Abstract). *Wilkerson* is not directed towards analyzing a set of time segments in a log to determine whether a time gap tolerance has been exceeded for a time segment. Instead, *Wilkerson* is directed toward a fully automated recovery system where users do not need to look up, write, or enter any program control language.

In contrast, *Wu* teaches mapping client access patterns in order to provide user-oriented object usage statistics, *Boyd* teaches efficiently analyzing server traffic through summarizing existing summaries while subtracting inflated counts and *Myerson* teaches supplementing a log

file to more accurately represent object requests made by a client. Thus, *Wilkerson* is directed towards an automated database recovery process for people not skilled in computer use, *Myerson* is directed towards supplementing log files with cached data, *Wu* is directed towards user-oriented object usage statistics, *Boyd* is directed towards efficiently analyzing server traffic and subtracting inflated counts, and none of the cited references is directed towards analyzing a set of time segments in a log to determine whether a time gap tolerance has been exceeded for a time segment, as is the present invention.

Additionally, there is no teaching or suggestion in the references as to the desirability of including the features from the other references. As the Examiner has failed to demonstrate any motivation or incentive in the prior art to combine and modify the references so as to achieve the claimed invention, the alleged combination can only be the result of impermissible hindsight reconstruction using Applicant's own disclosure as a guide. While Applicant understands that all examination entails some measure of hindsight, when the rejection is based completely on hindsight, as in the present case, to the exclusion of what can be gleaned from the references, then the rejection is improper and should be withdrawn.

Even if *Wu* was properly combinable with *Boyd*, *Myerson*, and *Wilkerson*, the result of such a combination would not be the invention as recited in claims 4, 10, 19, 25, 32 and 38 of the present invention. Rather, such an alleged combination would result in a system substantially as taught in *Wu* in addition to the features of summarizing data previously existing for time slices, adding data to log files when caching occurs, and an automated database recovery process for people who are not technically skilled in the use of a computer. Even considering *Wilkerson*, the cited references fail to teach or suggest analyzing a set of time segments in a log to determine whether a time gap tolerance has been exceeded for a time segment, as recited in the claims upon which claims 4, 10, 19, 25, 32 and 38 depend.

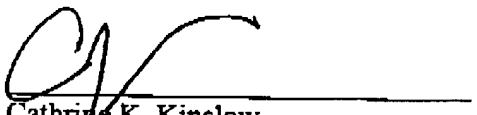
Furthermore, if an independent claim is nonobvious under 35 U.S.C. §103, then any claim depending therefrom is nonobvious. *In re Fine*, 837 F.2d 1071, 5 USPQ2d 1596 (Fed. Cir. 1988). Claims 4, 10, 19, 25, 32, and 38 are dependent claims depending upon independent claims 1, 8, 16, 23, 29, and 36, respectively. Applicant has already demonstrated claims 1, 8, 16, 23, 29, and 36 to be in condition for allowance. Applicant respectfully submits that claims 4, 10, 19, 25, 32, and 38 are also allowable, at least by virtue of their dependency on an allowable claim.

Therefore, the rejection of claims 4, 10, 19, 25, 32, and 38 under 35 U.S.C. § 103 has been overcome.

CONCLUSION

For the reasons stated above, Appellants respectfully submit that the rejection under 35 U.S.C. §103(a) of claims 1, 3-16, 18-29, and 31-41 has been overcome. Accordingly, Appellants respectfully request that the Board of Patent Appeals and Interferences overturn the rejections set forth in the Final Office Action.

Respectfully submitted,



Cathrine K. Kinslow
Reg. No. 51,886
YEE & ASSOCIATES, P.C.
PO Box 802333
Dallas, TX 75380
(972) 385-8777

CK/jl

CLAIMS APPENDIX

The text of the claims involved in the appeal are:

1. A method in a data processing system for maintaining data integrity in logs, the method comprising:

reviewing a log, wherein the log includes a set of time segments, and wherein the set of time segments include at least one time segment defined by a user;

determining whether the log contains a data loss, wherein the determining step includes analyzing each time segment within the set of time segments to determine whether a time segment gap tolerance has been exceeded; and

adding data to replace the data loss in the log to increase integrity of the log if a determination is made that a data loss has occurred.

3. The method of claim 1, wherein the data added to replace the data loss comprises data derived from a prior log.

4. The method of claim 1, wherein the data added to replace the data loss comprises data derived from a set of prior logs.

5. The method of claim 1, wherein the log includes data indicating at least one of requests, page views, and sessions.

6. The method of claim 1, wherein the analyzing step includes considering data in at least one time segment adjacent to a time segment being analyzed.

7. The method of claim 1, wherein the log is a Web server log.

8. A method in a data processing system for analyzing a log, the method comprising:

analyzing a set of time segments in the log to determine whether a time gap tolerance has been exceeded for a time segment within the set of time segments, wherein the set of time segments include at least one time segment defined by a user; and

responsive to a determination that the time gap tolerance has been exceeded for the time segment within the set of time segments, generating an alert.

9. The method of claim 8 further comprising:

responsive to detecting the alert, adding data to the time segment within the set of time segments to increase the data integrity of the log.

10. The method of claim 8, wherein the alert is presented on a user interface.

11. The method of claim 8, wherein the alert is a flag used by a program to process the log.

12. The method of claim 8 further comprising:

calculating a data integrity level for the log.

13. The method of claim 8, wherein the set of time segments include data for at least one of requests, page views, and sessions.

14. A data processing system comprising:

a bus system;

a communications unit connected to the bus system;

a memory connected to the bus system, wherein the memory includes a set of instructions; and

a processing unit connected to the bus system, wherein the processing unit executes the set of instructions to review a log, wherein the log includes a set of time segments, and wherein the set of time segments include at least one time segment defined by a user; determine whether the log contains a data loss by analyzing each time segment within the set of time segments to determine whether a time segment gap tolerance has been exceeded; and add data to replace the data loss in the log to increase integrity of the log if a determination is made that a data loss has occurred.

15. A data processing system comprising:

a bus system;

a communications unit connected to the bus system;

a memory connected to the bus system, wherein the memory includes a set of instructions; and

a processing unit connected to the bus system, wherein the processing unit executes the set of instructions to analyze a set of time segments in the log to determine whether a time gap

tolerance has been exceeded for a time segment within the set of time segments, wherein the set of time segments include at least one time segment defined by a user; and generate an alert in response to a determination that the time gap tolerance has been exceeded for the time segment within the set of time segments.

16. A data processing system for maintaining data integrity in logs, the data processing system comprising:

reviewing means for reviewing a log, wherein the log includes a set of time segments, and wherein the set of time segments include at least one time segment defined by a user;

determining means for determining whether the log contains a data loss by analyzing each time segment within the set of time segments to determine whether a time segment gap tolerance has been exceeded; and

adding means for adding data to replace the data loss in the log to increase integrity of the log if a determination is made that a data loss has occurred.

18. The data processing system of claim 16, wherein the data added to replace the data loss comprises data derived from a prior log.

19. The data processing system of claim 16, wherein the data added to replace the data loss comprises data derived from a set of prior logs.

20. The data processing system of claim 16, wherein the log includes data indicating at least one of requests, page views, and sessions.

21. The data processing system of claim 16, wherein the analyzing means includes considering data in at least one time segment adjacent to a time segment being analyzed.

22. The data processing system of claim 16, wherein the log is a Web server log.

23. A data processing system for analyzing a log, the data processing system comprising:

analyzing means for analyzing a set of time segments in the log to determine whether a time gap tolerance has been exceeded for a time segment within the set of time segments, wherein the set of time segments include at least one time segment defined by a user; and

generating means, responsive to a determination that the time gap tolerance has been exceeded for the time segment within the set of time segments, for generating an alert.

24. The data processing system of claim 23 further comprising:

adding means, responsive to detecting the alert, for adding data to the time segment within the set of time segments to increase the data integrity of the log.

25. The data processing system of claim 23, wherein the alert is presented on a user interface.

26. The data processing system of claim 23, wherein the alert is a flag used by a program to process the log.

27. The data processing system of claim 23 further comprising:

calculating means for calculating a data integrity level for the log.

28. The data processing system of claim 23, wherein the set of time segments include data for at least one of requests, page views, and sessions.

29. A computer program product in a computer readable medium for maintaining data integrity in logs, the computer program product comprising:

first instructions for reviewing a log, wherein the log includes a set of time segments, and wherein the set of time segments include at least one time segment defined by a user;

second instructions for determining whether the log contains a data loss by analyzing each time segment within the set of time segments to determine whether a time segment gap tolerance has been exceeded; and

third instructions for adding data to replace the data loss in the log to increase integrity of the log if a determination is made that a data loss has occurred.

31. The computer program product of claim 29, wherein the data added to replace the data loss comprises data derived from a prior log.

32. The computer program product of claim 29, wherein the data added to replace the data loss comprises data derived from a set of prior logs.

33. The computer program product of claim 29, wherein the log includes data indicating at least one of requests, page views, and sessions.

34. The computer program product of claim 29, wherein the sub-instructions includes considering data in at least one time segment adjacent to a time segment being analyzed.

35. The computer program product of claim 29, wherein the log is a Web server log.

36. A computer program product in a computer readable medium for analyzing a log, the computer program product comprising:

first instructions for analyzing a set of time segments in the log to determine whether a time gap tolerance has been exceeded for a time segment within the set of time segments, wherein the set of time segments include at least one time segment defined by a user; and

second instructions, responsive to a determination that the time gap tolerance has been exceeded for the time segment within the set of time segments, for generating an alert.

37. The computer program product of claim 36 further comprising:

third instructions, responsive to detecting the alert, for adding data to the time segment within the set of time segments to increase the data integrity of the log.

38. The computer program product of claim 36, wherein the alert is presented on a user interface.

39. The computer program product of claim 36, wherein the alert is a flag used by a program to process the log.

40. The computer program product of claim 36 further comprising:

third instructions for calculating a data integrity level for the log.

41. The computer program product of claim 36, wherein the set of time segments include data for at least one of requests, page views, and sessions.

EVIDENCE APPENDIX

There is no evidence to be presented.

RELATED PROCEEDINGS APPENDIX

There are no related proceedings.